MixGR: Enhancing Retriever Generalization for Scientific Domain through Complementary Granularity

Fengyu Cai[1] **Xinran Zhao**[2] **Tong Chen**[3] **Sihao Chen**[4]
Hongming Zhang[5] **Iryna Gurevych**[1] **Heinz Koeppl**[1]

[1]Technical University of Darmstadt [2]Carnegie Mellon University [3]University of Washington
[4]University of Pennsylvania [5]Tencent AI Lab
{fengyu.cai, heinz.koeppl}@tu-darmstadt.de

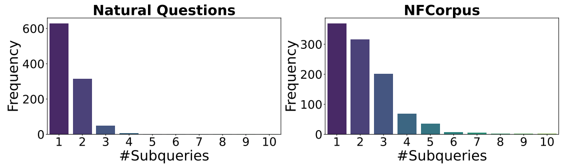
Abstract

Recent studies show the growing significance of document retrieval in the generation of LLMs, i.e., RAG, within the scientific domain by bridging their knowledge gap. However, dense retrievers often struggle with domain-specific retrieval and complex query-document relationships, particularly when query segments correspond to various parts of a document. To alleviate such prevalent challenges, this paper introduces MixGR, which improves dense retrievers' awareness of query-document matching across various levels of granularity in queries and documents using a zero-shot approach. MixGR fuses various metrics based on these granularities to a united score that reflects a comprehensive query-document similarity. Our experiments demonstrate that MixGR outperforms previous document retrieval by 24.7%, 9.8%, and 6.9% on nDCG@5 with unsupervised, supervised, and LLM-based retrievers, respectively, averaged on queries containing multiple subqueries from five scientific retrieval datasets. Moreover, the efficacy of two downstream scientific question-answering tasks highlights the advantage of MixGR to boost the application of LLMs in the scientific domain. The code and experimental datasets are available. [1]

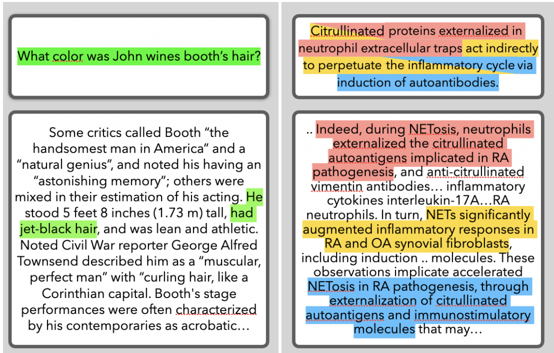
1 Introduction

Recent advances in Large Language Models (LLMs) have significantly impacted various scientific domains (Zhang et al., 2022; Touvron et al., 2023; Birhane et al., 2023; Grossmann et al., 2023). However, LLMs are notorious for their tendency to produce hallucinations, generating unreliable outputs (Ji et al., 2023). To address this, Retrieval-Augmented Generation (RAG; Lewis et al. 2020) has been developed to address this issue by incorporating external knowledge during the generation.

Though notable for accessing external and relevant knowledge, dense retrievers face specific challenges in the scientific domain: (1) *Domain-specific* nature: dense retrievers are typically trained on the general corpus such as Natural Questions (NQ; Kwiatkowski et al. 2019). However, scientific domains differ notably, e.g., the terminology and the pattern of queries as shown in Figure 1a. (2) *Complexity* of scientific documents: they are long, structured (Erera et al., 2019) and contain complex relationships between arguments (Kirschner et al., 2015). Figure 1a demonstrates that scientific queries tend to contain more subqueries than those in general domains. This indicates that subqueries within a single query may align with different parts



(a) Subquery distribution of general and scientific queries: scientific queries, e.g., NFCorpus (Boteva et al. 2016, *Right*), demonstrate a more diverse range of subqueries per query than general queries, e.g., Natural Questions (Kwiatkowski et al. 2019, *Left*).



(b) Comparison between general and scientific query-doc retrieval: compared with the general query-doc retrieval exemplified by NQ (Kwiatkowski et al. 2019, *Left*), the scientific query-doc retrieval exemplified by SciFact (Wadden et al. 2020, *Right*) demonstrates that one query can be decomposed to multiple subqueries, which can be mapped to different parts of documents.

Figure 1: Scientific document retrieval is shown to be more complicated than general domains.

[1]https://github.com/TRUMANCFY/MixGR

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 10369–10391

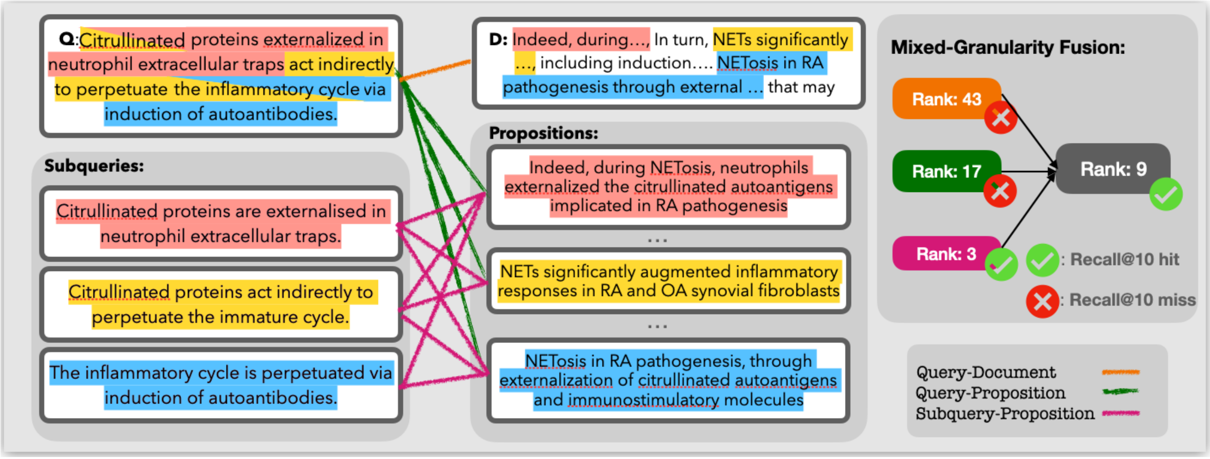


Figure 2: The illustration of `MixGR`: Both queries and documents (e.g., the query-doc pair from SciFact in Figure 1b) are decomposed into subqueries and propositions, respectively, each containing distinct semantic components. Starting from the original queries and documents along with their decomposed elements, metrics from various granularity combinations are fused into a single integrated score.

of a document (doc), resulting in complex interactions between queries and documents (Figure 1b). Such complexity poses significant challenges for dense retrievers (Lupart et al., 2023). Addressing these challenges requires specific training on the scientific corpus. However, this is often hindered by the necessity of extensive annotations (Wadden et al., 2020) and extra computation (Wang et al., 2021a).

In this study, we introduce a novel **zero-shot** approach that effectively adapts dense retrievers to scientific domains. This method specifically addresses the complexities arising from the composition of scientific queries and their consequent intricate relationships with documents. Inspired by Chen et al. (2023), showing that finer units improve retrievers' generalization to rare entities, we incorporate more granular retrieval units, specifically propositions (prop), to address domain-specific challenges as shown in Figure 2. Given the complexity between scientific queries and documents (Figure 1b), we also consider finer units within queries–subqueries–to measure query-doc similarity at a finer granularity. This metric captures the similarity between subqueries and propositions, moving beyond simple point similarity between query-doc vectors. Given a query, the distribution of corresponding information within a document is unknown. Additionally, our empirical analysis reveals that similarities at various granularities provide complementary insights. Therefore, for each query-doc pair, we fuse the metrics from these granularities to a unified score, termed **Mix**ed-**G**ranularity **R**etrieval as `MixGR`, as depicted in Figure 2.

We conducted document retrieval experiments on five scientific datasets using eight retrievers, comprising two unsupervised and four supervised, and two LLM-based models. Our results demonstrate that `MixGR` markedly surpasses previous query-doc retrieval methods. Notably, we recorded an average improvement of 24.7% for unsupervised retrievers, 9.8% for supervised ones, and 6.9% for LLM-based ones in terms of nDCG@5 for queries involving multiple subqueries. Furthermore, documents retrieved via `MixGR` substantially enhance the performance of downstream scientific QA tasks, underscoring their potential utility for RAG within scientific domains.

Our contributions are three-fold:

- We identify the challenges within scientific document retrieval, i.e., domain shift and query-doc complexity. We initiate retrieval with mixed granularity within queries and documents to address these issues;
- We propose `MixGR`, which further incorporates finer granularities within queries and documents, computes query-doc similarity over various granularity combinations and fuses them as a united score. Our experiments across five datasets and eight retrievers empirically reveal that `MixGR` significantly enhances existing retrievers on the scientific document retrieval and downstream QA tasks;
- Further analysis demonstrates the complementarity of metrics based on different granularities and

the generalization of MixGR in retrieving units finer than documents.

2 Preliminary and Related works

Generalization of Dense Retrievers Dense retrievers generally employ a dual-encoder framework (Yih et al., 2011; Reimers and Gurevych, 2019) to separately encode queries and documents into compact vectors and measure relevance using a non-parametric similarity function (Mussmann and Ermon, 2016). However, the simplicity of the similarity function (e.g., cosine similarity) can restrict expressiveness, leading to suboptimal generalization in new domains such as scientific fields that differ from original training data (Thakur et al., 2021). To improve dense retrievers' adaptability across tasks, researchers have used data augmentation (Wang et al., 2022; Lin et al., 2023; Dai et al., 2023), continual learning (Chang et al., 2020; Sachan et al., 2021; Oguz et al., 2022), and task-aware training (Xin et al., 2022; Cheng et al., 2023). However, these methods still require training on domain-specific data, incurring additional computational costs. This work focuses on *zero-shot* generalization of dense retrievers to scientific fields by incorporating multi-granularity similarities within queries and documents.

Granularity in Retrieval For dense retrieval, the selection of the retrieval unit needs to balance the trade-off between completeness and compactness. Coarser units, like documents or fixed-length passages, theoretically encompass more context but may introduce extraneous information, adversely affecting retrievers and downstream tasks (Shi et al., 2023; Wang et al., 2023). Conversely, finer units like sentences are not always self-contained and may lose context, thereby hindering retrieval (Akkalyoncu Yilmaz et al., 2019; Yang et al., 2020). Additionally, some studies extend beyond complete sentences; for example, Lee et al. (2021a) use phrases as learning units to develop corresponding representations. Meanwhile, ColBERT (Khattab and Zaharia, 2020) addresses token-level query-doc interaction but is hampered by low efficiency.

Chen et al. (2023) propose using *propositions* as retrieval units, defined as atomic expressions of meaning (Min et al., 2023). These units are contextualized and self-contained, including necessary context through decontextualization, e.g., coreference resolution (Zhang et al., 2021). Proposition retrieval improves retrieval of documents with long-tail information, potentially benefiting domain-specific tasks. This motivates the use of propositions as retrieval units for scientific document retrieval. Furthermore, we extend fine granularity to queries and enhance the query-doc similarity measurement, moving from a point-wise assessment between two vectors to integrating multiple query-doc granularity combinations.

Fusion within Retrieval Each type of retriever, sparse or dense, has its own strength and can be complementary with each other. Based on this insight, previous studies have explored the fusion of searches conducted by different retrievers as a zero-shot solution for domain adaptation (Thakur et al., 2021). A common method involves the convex combination, which linearly combines similarity scores (Karpukhin et al., 2020; Wang et al., 2021b; Ma et al., 2021). However, this approach is sensitive to the weighting of different metrics and score normalization, which complicates configuration across different setups (Chen et al., 2022).

In this work, we enhance retrieval by integrating searches across various query and document granularity levels for a given retriever. To avoid the limitations of convex combination on parameter searching, we use Reciprocal Rank Fusion (RRF; Cormack et al. 2009), a robust, non-parametric method (Chen et al., 2022), to aggregate these searches.

3 MixGR: Mix-Granularity Retrieval

3.1 Finer Units in Queries and Documents

We first decompose queries and documents into atomic units, i.e., subqueries and propositions, respectively. A proposition (or subquery) should meet the following three principal criteria (Min et al., 2023):

- Each proposition conveys a distinct semantic unit, collectively expressing the complete meaning.
- Propositions should be atomic and indivisible.
- According to Choi et al. (2021), propositions should be contextualized and self-contained, including all necessary text information such as resolved coreferences for clear interpretation.

Here, we employ an off-the-shelf model, *propositioner*,[2] for decomposing queries and documents (Chen et al., 2023). This model is developed by

[2] https://huggingface.co/chentong00/propositionizer-wiki-flan-t5-large

	Query	Document
Accuracy (%)	96.3	94.7
IAA (%)	92.0	89.0

Table 1: Human-evaluated accuracy of query/document decomposition by *propositioner* (Chen et al., 2023).

distilling the decomposition capacities of GPT-4 (Achiam et al., 2023) to a Flan-T5-Large model (Chung et al., 2024) using Wikipedia as the corpus. We sample decomposition results from 100 queries and 100 documents from the datasets in §4.1 and manually label the correctness of decomposition as shown in Table 1. This model is shown to effectively decompose queries and documents into atomic units within scientific domains. Please see Appendix B for further details.

3.2 Multi-Granularity Similarity Calculation

Given these various granularities including queries, subqueries, documents, and propositions, we extend the query-doc similarity metrics to include measurements across different combinations of granularities as depicted in Figure 2.

Notations The sets of queries and documents are denoted as $\mathcal{Q}$ and $\mathcal{D}$, respectively. Given a retriever s, the similarity between a query $\mathrm{q} \in \mathcal{Q}$ and a document $\mathrm{d} \in \mathcal{D}$ is denoted as $s(\mathrm{q}, \mathrm{d})$. A document d can be decomposed to N propositions, i.e., $\mathrm{d} = [d_1, ..., d_N]$. And a query q can be decomposed to M subqueries, i.e., $\mathrm{q} = [q_1, ..., q_M]$.

Query-doc $s_{q\text{-}d}$: The direct and original similarity between q and d is $s_{q\text{-}d}(\mathrm{q}, \mathrm{d}) \equiv s(\mathrm{q}, \mathrm{d})$.

Query-prop $s_{q\text{-}p}$: Recent works (Chen et al., 2023) determine query-doc similarity by calculating the maximum similarity between the **query** and individual **propositions** within the document (Lee et al., 2021b; Chen et al., 2023). The computation of this metric, denoted as $s_{q\text{-}p}$, is as follows:

$$s_{q\text{-}p}(\mathrm{q}, \mathrm{d}) = \max_{i=1,...,N} \{s(\mathrm{q}, d_i)\}. \quad (1)$$

Subquery-prop $s_{s\text{-}p}$: Considering that different parts of a query may be captured by various propositions within a document shown in Figure 1b, we further assess query-doc similarity by analyzing the relationships between **subqueries** and individual **propositions**. The similarity between a query and a document can be defined as the average similarity across subqueries, calculated by identifying the maximum similarity between one subquery and each proposition, in analogy to MaxSim in ColBERT (Khattab and Zaharia, 2020). This metric, represented by $s_{s\text{-}p}$, is calculated as:

$$s_{s\text{-}p}(\mathrm{q}, \mathrm{d}) = \frac{1}{M} \sum_{i=1}^{M} \max_{j=1,...,N} \{s(q_i, d_j)\}. \quad (2)$$

3.3 Reciprocal Rank Fusion

We then use RRF to fuse these metrics across different query and document granularities:

$$s_f(\mathrm{q}, \mathrm{d}) = \frac{1}{1 + r_{q\text{-}d}(\mathrm{q}, \mathrm{d})} + \frac{1}{1 + r_{q\text{-}p}(\mathrm{q}, \mathrm{d})} + \frac{1}{1 + r_{s\text{-}p}(\mathrm{q}, \mathrm{d})}, \quad (3)$$

where $r_{q\text{-}d}$, $r_{q\text{-}p}$, $r_{s\text{-}p} \in \mathbb{R}_{\geq 0}$ signify the rank of the retrieve results by $s_{q\text{-}d}$, $s_{q\text{-}p}$, and $s_{s\text{-}p}$, respectively. Technically, we retrieve the top-k results $R^k_{q\text{-}d}$, $R^k_{q\text{-}p}$, and $R^k_{s\text{-}p}$ by $s_{q\text{-}d}$, $s_{q\text{-}p}$, and $s_{s\text{-}p}$, respectively, where k is set 200 empirically. When a query-doc pair (q', d') in one retrieval result does not exist in the other sets (e.g., $(\mathrm{q}', \mathrm{d}') \in R^k_{q\text{-}d}$ but $(\mathrm{q}', \mathrm{d}') \notin R^k_{q\text{-}p}$), we will calculate the missing similarity (e.g., $s_{q\text{-}p}(\mathrm{q}', \mathrm{d}')$) before aggregation.

4 Experimental Setting

4.1 Scientific Retrieval Datasets

We evaluate our approach on five different scientific retrieval tasks, including BioASQ (Tsatsaronis et al., 2015), NFCorpus (Boteva et al., 2016), SciDocs (Cohan et al., 2020), SciFact (Wadden et al., 2020), and SciQ (Welbl et al., 2017), as shown in Table 4 in Appendix A. We employ the *propositioner* released by Chen et al. (2023) mentioned in §3.1 to break down both queries and documents into atomic units. As we focus with priority on query-doc complexity in scientific domains, we report the experiments and analysis on the subset of the queries that contain multiple subqueries. For queries containing only a single subquery, we also assess the methodology of `MixGR` by omitting $s_{s\text{-}p}$. The detailed results are shown in Table 9 in Appendix G.4. The results encompassing all queries, ones containing both single and multiple subqueries, are detailed in Table 9 located in Appendix G.4.

4.2 Dense Retrievers

We evaluate the performance of eight off-the-shelf dense retrievers, both supervised and unsupervised

Retriever	Setup	BioASQ		NFCorpus		SciDocs		SciFact		SciQ		Avg.	
		ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20
						Unsupervised Dense Retrievers							
SimCSE	$s_{q\text{-}d}$	17.0	17.0	16.2	13.3	7.6	9.7	27.1	31.2	62.3	67.3	26.0	27.7
	$s_{q\text{-}p}$	28.4	28.2	20.0	16.4	8.2	11.1	32.8	37.2	75.6	78.5	33.0	34.3
	$s_{s\text{-}p}$	**31.1**	30.6	**22.8**	**18.3**	7.3	10.5	32.7	36.9	80.9	83.2	35.0	35.9
	MixGR	30.7	**31.3**	22.3	18.1	**9.1**	**12.2**	**34.8**	**39.8**	**84.0**	**85.5**	**36.2** +39.2%	**37.4** +35.0%
Contriever	$s_{q\text{-}d}$	64.8	68.3	42.2	34.9	13.5	18.5	64.5	68.5	67.2	70.0	50.5	52.0
	$s_{q\text{-}p}$	64.1	68.2	43.0	35.5	14.5	19.4	64.0	68.9	79.7	81.0	53.1	54.6
	$s_{s\text{-}p}$	63.7	68.0	41.4	34.9	13.5	18.3	63.2	67.5	83.6	84.6	53.1	54.6
	MixGR	**67.0**	**71.7**	**44.0**	**37.1**	**15.5**	**20.7**	**66.4**	**71.0**	**85.2**	**86.7**	**55.6** +10.1%	**57.5** (+10.6%)
						Supervised Dense Retrievers							
DPR	$s_{q\text{-}d}$	39.1	39.1	25.1	20.7	7.3	10.4	31.8	37.7	60.6	64.1	32.8	34.4
	$s_{q\text{-}p}$	43.1	43.8	25.2	20.6	7.8	10.6	36.1	40.5	63.6	67.9	35.2	36.7
	$s_{s\text{-}p}$	41.1	41.7	26.5	21.4	6.4	10.0	37.1	41.3	67.7	70.7	35.8	37.0
	MixGR	**44.6**	**45.8**	**27.7**	**22.9**	**8.2**	**11.5**	**39.4**	**43.6**	**73.6**	**76.1**	**38.7** +18.0%	**40.0** +14.3%
ANCE	$s_{q\text{-}d}$	48.8	48.2	29.9	24.4	9.3	13.1	41.5	45.3	66.4	69.1	39.2	40.0
	$s_{q\text{-}p}$	53.0	53.7	29.4	24.0	9.2	12.9	43.3	46.4	62.3	66.4	39.4	40.7
	$s_{s\text{-}p}$	49.0	49.8	30.3	24.5	7.5	11.9	43.5	47.3	66.1	69.1	39.3	40.5
	MixGR	**53.4**	**54.7**	**31.9**	**25.9**	**9.6**	**14.1**	**46.8**	**49.9**	**74.4**	**76.8**	**43.2** +10.2%	**44.3** +10.4%
TAS-B	$s_{q\text{-}d}$	68.8	70.1	42.3	34.1	13.8	19.3	60.1	65.6	84.8	86.3	54.0	55.1
	$s_{q\text{-}p}$	70.9	72.3	42.5	34.4	**14.3**	18.1	60.7	64.4	85.6	86.3	54.8	55.1
	$s_{s\text{-}p}$	67.0	69.4	40.9	33.1	12.6	17.2	61.7	65.0	85.3	86.6	53.5	54.3
	MixGR	**71.7**	**74.0**	**43.6**	**35.2**	14.0	**19.6**	**62.7**	**66.9**	**90.5**	**91.0**	**56.5** +4.6%	**57.3** +4.0%
GTR	$s_{q\text{-}d}$	63.5	63.1	42.1	34.1	**13.6**	18.9	58.3	62.2	83.3	84.4	52.2	52.5
	$s_{q\text{-}p}$	65.9	67.8	42.3	34.4	13.2	18.0	60.6	63.3	85.8	86.5	53.6	54.0
	$s_{s\text{-}p}$	61.2	63.5	41.5	33.6	11.6	16.2	58.4	62.0	88.5	89.0	52.3	52.9
	MixGR	**66.8**	**68.6**	**43.3**	**35.6**	**13.6**	**19.2**	**60.9**	**64.5**	**92.9**	**93.0**	**55.5** +6.3%	**56.2** +7.0%
						LLM-based Retrievers							
GTE-Qwen	$s_{q\text{-}d}$	54.8	56.2	40.2	32.9	12.2	16.9	62.9	67.5	73.0	75.1	48.6	49.7
	$s_{q\text{-}p}$	**69.5**	**73.1**	**45.5**	**37.6**	**17.7**	**24.9**	69.9	73.9	81.6	83.1	56.9	58.5
	$s_{s\text{-}p}$	61.5	66.0	43.7	35.8	15.1	21.9	67.5	71.2	82.1	83.6	54.0	55.7
	MixGR	66.5	70.1	44.9	37.5	17.2	23.5	**70.3**	**74.3**	**86.7**	**87.5**	**57.1** +11.5%	**58.6** +17.7%
E5-Mistral	$s_{q\text{-}d}$	73.2	76.4	50.3	41.7	**19.1**	**25.5**	74.6	77.4	87.5	88.4	60.9	61.9
	$s_{q\text{-}p}$	**76.8**	**80.3**	49.7	41.4	17.5	23.7	76.0	78.6	82.0	84.1	60.4	61.6
	$s_{s\text{-}p}$	70.9	75.6	47.3	39.1	15.8	21.2	73.2	76.4	84.3	85.5	58.3	59.6
	MixGR	75.9	79.8	**50.4**	**42.0**	18.1	24.9	**76.5**	**79.2**	**90.7**	**91.2**	**62.3** +2.3%	**63.4** +2.4%

Table 2: Document Retrieval Performance (nDCG@k = 5, 20 in percentage, abbreviated as ND@k): We evaluated five distinct scientific retrieval datasets using two unsupervised, four supervised, and two LLM-based retrievers. The retrieval results were compared among various metrics: $s_{q\text{-}d}$ (previous query-doc similarity), $s_{q\text{-}p}$ (Chen et al., 2023), $s_{s\text{-}p}$, and MixGR, as detailed in §3.2. **Bold** presents the best performance across the metrics, while underline denotes the second-best performance. MixGR outperforms all three other metrics, where the percentage in parentheses indicates the relative improvement compared with $s_{q\text{-}d}$.

ones together with LLM-based models. [3] Supervised retrievers are trained using human-labeled query-doc pairs in general domains,[4] while unsupervised models do not require labeled data. LLM-based retrievers utilize a decoder-only LLM as the core model, encoding both queries and documents using a specially designed token. These retrievers encode the queries and index the corpus at both document and proposition levels:

- SimCSE (Gao et al., 2021) employs a BERT-base (Devlin et al., 2019) encoder trained on randomly selected unlabeled Wikipedia sentences.
- Contriever (Izacard et al., 2022) is an unsupervised retriever evolved from a BERT-base encoder, contrastively trained on segments from unlabelled web and Wikipedia documents.
- DPR (Karpukhin et al., 2020) is built with a dual-encoder BERT-base architecture, finetuned on a suite of open-domain datasets with labels, such as SQuAD (Rajpurkar et al., 2016).
- ANCE (Xiong et al., 2021) mirrors the configuration of DPR but incorporates a training scheme of Approximate Nearest Neighbor Negative Contrastive Estimation (ANCE).
- TAS-B (Hofstätter et al., 2021) is a dual-encoder BERT-base model distilled from ColBERT on MS MARCO (Bajaj et al., 2016).
- GTR (Ni et al., 2022) is a T5-base encoder, focus-

[3] Though LLM-based retrievers are also trained using query-doc pairs, their significantly large model sizes set them apart as a distinct category.

[4] The supervised retrievers used in our experiment have not been trained on these five datasets.

ing on generalization, pre-trained on unlabeled QA pairs, and fine-tuned on labeled data including MS MARCO.

- GTE-Qwen2-1.5B-instruct (GTE-Qwen, Li et al. 2023b) is built on Qwen2-1.5B (Yang et al., 2024) and trained with multi-stage contrastive learning over a diverse mixture of datasets.
- E5-Mistral-7b-instruct [5] (E5-Mistral, Wang et al. 2024) is initialized from Mistral-7B-v0.1 (Jiang et al., 2023) and finetuned on a set of multilingual datasets. [EOS] is appended to both the query and document before being fed into a pre-trained LLM to obtain embeddings from the last vector, and the model is trained using standard InfoNCE loss (Radford et al., 2021) with in-batch and hard negatives.

More details on retrievers and experimental setups are presented in Appendices C and D.

4.3 Document Retrieval Evaluation

We assess the performance of `MixGR` in the task of document retrieval. Due to input length limitations for retrievers (Karpukhin et al., 2020), we divide each document into fixed-length chunks of up to 128 words. In practice, for `MixGR` and baselines, we identify the retrieved chunks, map them back to their original documents, and return the top-k documents. We use Normalised Cumulative Discount Gain (nDCG@k) as the evaluation metrics for document retrieval. Unlike Recall@k, which only indicates the presence of golden documents in the retrieved list, nDCG@k also accounts for both the ranking of retrievals and the relevance judgment of golden documents (Thakur et al., 2021). The baselines will be the metrics containing the homogeneous granularity introduced in the previous section, i.e., $s_{q\text{-}d}$, $s_{q\text{-}p}$ and $s_{s\text{-}p}$.

4.4 Downstream QA Evaluation

As previously mentioned, scientific documents are vital for LLMs due to the rapid advancements in science and the limited availability of such content in training datasets. To better understand how `MixGR` enhances downstream QA tasks, we implement the *retrieval-then-read* approach on two datasets SciQ and SciFact. We retrieve and rank the top-k documents based on scores, $s_{q\text{-}d}$ and `MixGR`, then concatenate them to form the context. During our evaluations, we limit the number of document chunks retrieved to 1 and 3—thus, only the top k documents are injected into the reader model. We assess the performance by measuring the Exact Match (EM) rate—the proportion of responses where the predicted answer perfectly aligns with the reference answer (Kamalloo et al., 2023), denoted as EM@k. Specifically, we utilize LLama-3-8B-Instruct [6] (Touvron et al., 2023) as the reader model. We take the original query-doc retrieval setup, i.e., retrieval based on $s_{q\text{-}d}$, as the baseline. Please refer to Appendix F for more details.

5 Results

This section analyzes the impact of mixed-granularity retrieval on document retrieval and downstream applications. We highlight the effectiveness of our proposed fine-grained and mixed-granularity approaches in enhancing performance across various metrics.

5.1 Document Retrieval

Table 2 reports the results of document retrieval. We observe that retrieval by `MixGR` outperforms all single-granularity retrieval with both unsupervised and supervised dense retrievers in most cases.

With unsupervised retrievers, `MixGR` significantly outperforms the query-doc similarity, $s_{q\text{-}d}$, across all five datasets. There is an average nDCG@5 improvement of $+10.2$ and $+5.1$ (39.2% and 10.1% relatively) for SimCSE and Contriever, respectively.

With supervised and LLM-based retrievers, improvements associated with `MixGR` are also observed, although they are not as significant as with unsupervised retrievers. This indicates that `MixGR` effectively narrows the distributional gap between dense retrievers and scientific domains.

Unsupervised retrievers benefit more from `MixGR` than supervised ones. Remarkably, with `MixGR`, the unsupervised retriever Contriever outperforms supervised models, as evidenced by its superior average results across five datasets on nCCG@20. This result is particularly significant given that Contriever typically underperforms compared to TAS-B and GTR when evaluated using traditional query-document similarity measures. Additionally, the study (Thakur et al., 2021) reveals that sparse retrievers like BM25 often excel over

[5] Considering the computational time, we set the threshold k in §3.3 as 50 on BioASQ with E5-Mistral-7B-instruct.

[6] `https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct`

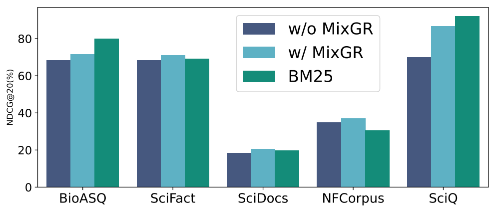


Figure 3: Comparison between BM25 and Contriever (w/ and w/o `MixGR`) on nDCG@20: Contriever w/ `MixGR` outperforms BM25 in three out of five datasets.

dense retrievers in domain-specific retrieval tasks. As shown in Figure 3, Contriever outperforms BM25 in three out of five datasets when applied with `MixGR`. These findings emphasize the substantial enhancements that `MixGR` contributes to unsupervised retrievers within scientific domains.

Finer granularity helps retrieval more. Among three metrics within `MixGR`, the query-proposition measurement $s_{q\text{-}p}$ and the subquery-proposition measurement $s_{s\text{-}p}$ show a general advantage over the original query-doc similarity, as highlighted by the underlined results in Table 2. The original query-doc metric, $s_{q\text{-}d}$, outperforms the subquery-proposition measurement only when using the retriever TAS-B and E5-Mistral. These findings corroborate and expand upon Chen et al. (2023), suggesting that finer query-doc similarity measurement significantly improves document retrieval performance.

5.2 Downstream QA Tasks

Table 3 reports the results of scientific question answering when the documents retrieved by `MixGR` are fed into LLMs, i.e. the readers. It is observed that EM scores achieved with `MixGR` generally surpass those of the baseline across two datasets, eight dense retrievers, and multiple numbers of input documents. This underscores the effectiveness of `MixGR` in enhancing the performance of downstream QA tasks.

6 Analysis

In this section, we explore the complementary advantages of various similarity metrics across multiple granularities within `MixGR` through an ablation study. Although the finer-granularity metric, $s_{s\text{-}p}$, generally enhances performance as previously discussed, it can occasionally result in degradation when compared to original query-document similarity $s_{q\text{-}d}$. We identify specific conditions under

	Setup	SciFact EM@1	SciFact EM@3	SciQ EM@1	SciQ EM@3
Unsupervised Dense Retrievers					
SimCSE	$s_{q\text{-}d}$	**50.0**$_{\pm 1.9}$	60.5$_{\pm 2.1}$	55.2$_{\pm 0.7}$	57.9$_{\pm 0.2}$
	`MixGR`	48.7$_{\pm 1.8}$	**64.7**$_{\pm 1.7}$	**61.7**$_{\pm 0.9}$	**66.8**$_{\pm 0.6}$
Contriever	$s_{q\text{-}d}$	60.5$_{\pm 2.9}$	**71.9**$_{\pm 2.0}$	54.4$_{\pm 0.5}$	63.2$_{\pm 0.7}$
	`MixGR`	**61.8**$_{\pm 2.8}$	70.5$_{\pm 2.4}$	**61.7**$_{\pm 0.7}$	**66.1**$_{\pm 0.2}$
Supervised Dense Retrievers					
DPR	$s_{q\text{-}d}$	50.2$_{\pm 1.9}$	58.9$_{\pm 2.3}$	51.7$_{\pm 0.2}$	57.3$_{\pm 0.2}$
	`MixGR`	**51.1**$_{\pm 1.2}$	**65.1**$_{\pm 2.5}$	**57.5**$_{\pm 0.4}$	**62.5**$_{\pm 0.7}$
ANCE	$s_{q\text{-}d}$	50.8$_{\pm 2.0}$	63.4$_{\pm 2.6}$	53.0$_{\pm 0.2}$	59.1$_{\pm 0.4}$
	`MixGR`	**56.8**$_{\pm 2.6}$	**70.4**$_{\pm 2.1}$	**55.4**$_{\pm 0.2}$	**63.3**$_{\pm 0.3}$
TAS-B	$s_{q\text{-}d}$	**60.3**$_{\pm 4.6}$	**73.3**$_{\pm 1.4}$	60.8$_{\pm 0.5}$	66.3$_{\pm 0.4}$
	`MixGR`	59.5$_{\pm 3.4}$	69.9$_{\pm 2.7}$	**64.8**$_{\pm 0.6}$	**67.9**$_{\pm 0.4}$
GTR	$s_{q\text{-}d}$	58.4$_{\pm 3.4}$	70.0$_{\pm 2.5}$	60.0$_{\pm 0.5}$	65.1$_{\pm 0.5}$
	`MixGR`	**60.6**$_{\pm 2.8}$	**73.3**$_{\pm 2.1}$	**64.5**$_{\pm 0.6}$	**66.7**$_{\pm 0.2}$
LLM-based Retrievers					
GTE-Qwen	$s_{q\text{-}d}$	61.8$_{\pm 6.0}$	59.9$_{\pm 4.7}$	56.2$_{\pm 0.3}$	64.1$_{\pm 0.3}$
	`MixGR`	**63.7**$_{\pm 3.4}$	**68.2**$_{\pm 4.0}$	**62.6**$_{\pm 0.8}$	**68.0**$_{\pm 0.0}$
E5-Mistral	$s_{q\text{-}d}$	**64.0**$_{\pm 2.2}$	**78.1**$_{\pm 1.5}$	59.8$_{\pm 0.6}$	64.6$_{\pm 0.3}$
	`MixGR`	63.7$_{\pm 3.4}$	68.2$_{\pm 4.0}$	**62.6**$_{\pm 0.8}$	**68.0**$_{\pm 0.0}$

Table 3: Scientific Question Answering on SciFact and SciQ using Llama-3-8B-Instruct (Touvron et al., 2023): the top-1 and 3 document chunks retrieved by retrievers, following the metrics $s_{q\text{-}d}$ and `MixGR`, were fed into the reader. **Bold** indicates the better performance. Numbers following $\pm$ present standard deviations of EM, generated with a temperature of 0.1 and a top_p value of 0.7 across three seeds.

which the finer-granularity metric offers greater benefits. Previous works (Chen et al., 2023) primarily explored multiple granularities in *documents*. We conduct a controlled experiment to highlight the significance of incorporating multiple granularities in *queries* in the `MixGR` framework, which also validate the generalization of `MixGR` on the retrieval units finer than documents.

6.1 Ablation Study

In our ablation study, we conducted a systematic evaluation of the impact of various granularity measures—$s_{q\text{-}d}$ (query-doc similarity), $s_{q\text{-}p}$ (query-prop similarity), and $s_{s\text{-}p}$ (subquery-prop similarity)—on the performance of eight retrievers. By individually omitting each of these measures from the calculation of `MixGR` as defined in Equation 3, we assessed the significance of each granularity level. Specifically, the extent of performance degradation upon removal of a measure indicates its importance; greater degradation suggests higher importance of that particular granularity metric.

As illustrated in Figure 4, the nDCG@20 performance declined across all three setups and five datasets, demonstrating that the metrics are comple-

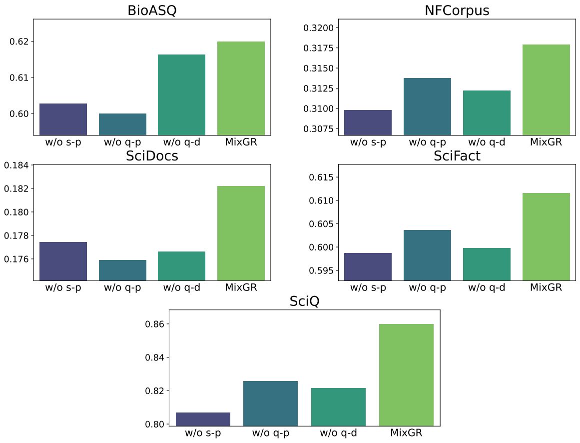


Figure 4: Ablation study of `MixGR` on the nDCG@20 metrics averaged on eight retrievers: `MixGR` achieves optimal performance when combining these three metrics, indicating their complementary nature.

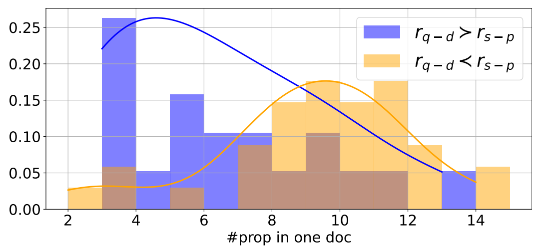


Figure 5: Distribution of proposition number within documents in two sets. There are more propositions within document when $r_{q\text{-}d} \prec r_{s\text{-}p}$ than $r_{q\text{-}d} \succ r_{s\text{-}p}$.

mentary to each other. The degree of performance degradation varied across different configurations, highlighting the importance of each granularity measure. Please refer to Table 7 in Appendix G.1 for detailed results.

6.2 When is finer granularity beneficial?

Therefore, to more effectively compare the impacts of $s_{q\text{-}d}$ and $s_{s\text{-}p}$, we categorized the *correctly* retrieved pairs (complex query, [7] doc) by `MixGR` in SciFact, using SimCSE, into two distinct groups:

- $r_{q\text{-}d} \succ r_{s\text{-}p}$: The query-doc rank of $s_{q\text{-}d}$ is higher than the subquery-prop rank of $s_{s\text{-}p}$;
- $r_{q\text{-}d} \prec r_{s\text{-}p}$: The query-doc rank of $s_{q\text{-}d}$ is lower than the subquery-prop rank of $s_{s\text{-}p}$.

Upon analyzing the number of propositions in documents, a significant pattern emerges: based on the distributions present in Figure 5, the number of propositions in $r_{q\text{-}d} \prec r_{s\text{-}p}$ is generally higher than

[7]We refer *complex query* as the query containing no fewer than three subqueries.

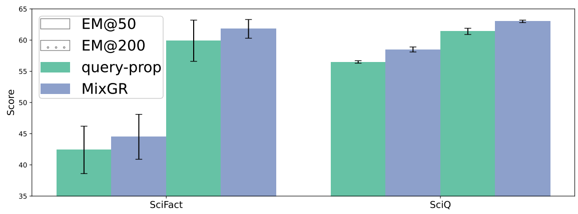


Figure 6: Proposition retrieval with `MixGR`: We evaluate Exact Match of LLama-3-8B-Instruct on SciFact and SciQ with the first 50 and 200 words of propositions, i.e., EM@50 and EM@200, retrieved by SimCSE as the context. The model generated results using a temperature of 0.1 and a top_p value of 0.7 across three different seeds. Please refer to Table 8 for other retrievers in Appendix G.2.

in $r_{q\text{-}d} \succ r_{s\text{-}p}$. This underscores the importance of incorporating finer units within documents, especially for those containing more propositions, and suggests potential degradation in dense retrievers when handling such documents. For other retrievers' results, please refer to Appendix G.3.

6.3 `MixGR` on Proposition Retrieval

Previous sections present the effectiveness of `MixGR` on scientific document retrieval. While previous works (Chen et al., 2023) focus on finer document granularity, we specifically assess `MixGR` on the proposition as the retrieval units. This controlled study highlights the benefits of `MixGR`, which incorporates different granularities within queries and documents, in general text retrieval beyond document-level granularity.

For a given query q and a proposition p, the conventional similarity is denoted by $s^p_{q\text{-}p} \equiv s(\mathrm{q}, \mathrm{p})$. When the query is further broken down into multiple sub-queries, we introduce a finer granularity measure, $s^p_{s\text{-}p}$, which is defined as the maximum similarity between these sub-queries and the proposition. $s^p_{s\text{-}p}$ is mathematically defined as follows:

$$s^p_{s\text{-}p}(\mathrm{q}, \mathrm{p}) = \max_{i=1,\ldots,M} \{s(q_i, \mathrm{p})\}. \tag{4}$$

Therefore, the merged score by RRF, $s^p_f(\mathrm{q}, \mathrm{p})$, is calculated as:

$$s^p_f(\mathrm{q}, \mathrm{p}) = \frac{1}{1 + r^p_{q\text{-}p}(\mathrm{q}, \mathrm{p})} + \frac{1}{1 + r^p_{s\text{-}p}(\mathrm{q}, \mathrm{p})}, \tag{5}$$

where $r^p_{q\text{-}p}$ and $r^p_{s\text{-}p}$ signify the rank of the retrieve results by $s^p_{q\text{-}p}$ and $s^p_{s\text{-}p}$, respectively.

Following $s^p_{q\text{-}p}(\mathrm{q}, \mathrm{p})$ and $s^p_f(\mathrm{q}, \mathrm{p})$, we input the first 50 and 200 words in propositions retrieved with SimCSE on SciFact and SciQ into the reader LLama-3-8B-Instruct. This process adheres to the same setups outlined in §4.4. As shown in Figure 6, the performance advance observed with mixed-granularity retrieval on propositions, compared to the original query-prop similarity, demonstrates the effectiveness of using mixed-granularity in retrieval. This substantiates the generalizability of MixGR beyond document-level granularity. Please refer to Appendix G.2 for details.

6.4 Prospect: Adaptive MixGR

Here, we outline potential future research directions. In §6.1, we observed the complementary nature of retrieval results achieved using different granularities. Additionally, as discussed in §6.2, we noted a distinct pattern where retrieval guided by a specific granularity outperforms others. These findings indicate that metrics based on different granularities each have relatively distinct strengths in specific contexts, presenting opportunities for further exploration. Unlike the non-parametric method of fusion by RRF, which overlooks the relative importance of components, an adaptive approach could enhance fusion and, consequently, improve retrieval performance with dense retrievers–a prospect we aim to explore in future research.

6.5 Efficiency Analysis

As one query may contain multiple subqueries, searching more queries will introduce an extra computational cost. For a query comprising M subqueries and a document containing N propositions, we denote the original query-document searching time as T. The experimental results in Table 6 in Appendix E demonstrate that the searching time for subquery-proposition retrieval will be approximately increased to MT. We expect that MixGR can be further optimized with more efficient searching algorithms, e.g., Tree-based index (Li et al., 2023a).

7 Conclusion

In this work, we identify key challenges for dense retrievers in scientific document retrieval, namely domain shift and query-document complexity. In response, we propose a zero-shot approach, MixGR, that utilizes atomic components in queries and documents to calculate their similarity with greater nuance. We then use Reciprocal Rank Fusion (RRF) to integrate these metrics, modeling query-doc similarity at different granularities into a unified score that enhances document retrieval.

Our experiments demonstrate that MixGR significantly enhances variant types of existing dense retrievers on document retrieval within the scientific domain. Moreover, MixGR has proven beneficial for downstream applications such as scientific QA. The analysis reveals a synergistic relationship among the components of MixGR, and suggests evolving our non-parametric fusion framework into a more general method as a future research direction.

Limitations

Our work explores retrieval guided by an integral metric that incorporates various levels of granularity. We identify several limitations in our approach: (1) *Coverage of Retrievers*: Our study categorizes dense retrievers into supervised, unsupervised, and LLM-based models, yet all utilize a dual-encoder structure. Future studies could include a more diverse array of retriever architectures. (2) *Coverage of Domains*: While our main focus is on the scientific domain, and we extend to three additional domains in Appendix H, there are still many domains we have not explored. (3) *Languages*: Our research is limited to an English corpus. The applicability of MixGR in multilingual contexts also deserves further validation and exploration. (4) Based on our error analysis in Appendix B.2, there is potential for improving the tool used for query and corpus decomposition.

Ethical Statements

We foresee no ethical concerns and potential risks in our work. All of the retrieval models and datasets are open-sourced, as shown in Table 12 in Appendix I. The LLMs we applied in the experiments are also publicly available. Given our context, the outputs of LLMs are unlikely to contain harmful and dangerous information.

Acknowledgement

We thank Kexin Wang, Tim Baumgärtner, and Sherry Tongshuang Wu for the discussion on an early draft of this work. We also thank anonymous reviewers for their suggestions during the review process.

Fengyu Cai is funded by the German Federal Ministry of Education and Research and the Hessian Ministry of Higher Education, Research, Science, and the Arts within their joint support of the National Research Center for Applied Cybersecurity ATHENE. Xinran Zhao is funded by the ONR Award N000142312840.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Zeynep Akkalyoncu Yilmaz, Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Cross-domain modeling of sentence-level evidence for document retrieval. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3490–3496, Hong Kong, China. Association for Computational Linguistics.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Abeba Birhane, Atoosa Kasirzadeh, David Leslie, and Sandra Wachter. 2023. Science in the age of large language models. *Nature Reviews Physics*, 5(5):277–280.

Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. 2016. A full-text learning to rank dataset for medical information retrieval. In *Advances in Information Retrieval: 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20–23, 2016. Proceedings 38*, pages 716–722. Springer.

Wei-Cheng Chang, X Yu Felix, Yin-Wen Chang, Yiming Yang, and Sanjiv Kumar. 2020. Pre-training tasks for embedding-based large-scale retrieval. In *International Conference on Learning Representations*.

Tao Chen, Mingyang Zhang, Jing Lu, Michael Bendersky, and Marc Najork. 2022. Out-of-domain semantics to the rescue! zero-shot hybrid retrieval models. In *European Conference on Information Retrieval*, pages 95–110. Springer.

Tong Chen, Hongwei Wang, Sihao Chen, Wenhao Yu, Kaixin Ma, Xinran Zhao, Dong Yu, and Hongming Zhang. 2023. Dense x retrieval: What retrieval granularity should we use? *arXiv preprint arXiv:2312.06648*.

Hao Cheng, Hao Fang, Xiaodong Liu, and Jianfeng Gao. 2023. Task-aware specialization for efficient and robust dense retrieval for open-domain question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1864–1875, Toronto, Canada. Association for Computational Linguistics.

Eunsol Choi, Jennimaria Palomaki, Matthew Lamm, Tom Kwiatkowski, Dipanjan Das, and Michael Collins. 2021. Decontextualization: Making sentences stand-alone. *Transactions of the Association for Computational Linguistics*, 9:447–461.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. 2020. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online. Association for Computational Linguistics.

Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 758–759.

Zhuyun Dai, Vincent Y Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith Hall, and Ming-Wei Chang. 2023. Promptagator: Few-shot dense retrieval from 8 examples. In *The Eleventh International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Shai Erera, Michal Shmueli-Scheuer, Guy Feigenblat, Ora Peled Nakash, Odellia Boni, Haggai Roitman, Doron Cohen, Bar Weiner, Yosi Mass, Or Rivlin, Guy Lev, Achiya Jerbi, Jonathan Herzig, Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, Francesca Bonin, and David Konopnicki. 2019. A summarization system for scientific documents. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 211–216, Hong Kong, China. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference* *on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Igor Grossmann, Matthew Feinberg, Dawn C Parker, Nicholas A Christakis, Philip E Tetlock, and William A Cunningham. 2023. Ai and the transformation of social science research. *Science*, 380:1108–1109.

Charles R. Harris, K. Jarrod Millman, Stéfan van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. 2020. Array programming with numpy. *Nature*, 585:357–362.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 113–122.

John D Hunter. 2007. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(03):90–95.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Ehsan Kamalloo, Nouha Dziri, Charles Clarke, and Davood Rafiei. 2023. Evaluating open-domain question answering in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5591–5606, Toronto, Canada. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

Christian Kirschner, Judith Eckle-Kohler, and Iryna Gurevych. 2015. Linking the thoughts: Analysis of argumentation structures in scientific publications. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 1–11, Denver, CO. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Jinhyuk Lee, Mujeen Sung, Jaewoo Kang, and Danqi Chen. 2021a. Learning dense representations of phrases at scale. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6634–6647, Online. Association for Computational Linguistics.

Jinhyuk Lee, Alexander Wettig, and Danqi Chen. 2021b. Phrase retrieval learns passage retrieval, too. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3661–3672, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Haitao Li, Qingyao Ai, Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Zheng Liu, and Zhao Cao. 2023a. Constructing tree-based index for efficient and effective dense retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 131–140.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023b. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Sheng-Chieh Lin, Akari Asai, Minghan Li, Barlas Oguz, Jimmy Lin, Yashar Mehdad, Wen-tau Yih, and Xilun Chen. 2023. How to train your dragon: Diverse augmentation towards generalizable dense retrieval. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6385–6400, Singapore. Association for Computational Linguistics.

Simon Lupart, Thibault Formal, and Stéphane Clinchant. 2023. Ms-shift: An analysis of ms marco distribution shifts on neural retrieval. In *European Conference on Information Retrieval*, pages 636–652. Springer.

Xueguang Ma, Kai Sun, Ronak Pradeep, and Jimmy Lin. 2021. A replication study of dense passage retriever. *arXiv preprint arXiv:2104.05740*.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: financial opinion mining and question answering. In *Companion proceedings of the the web conference 2018*, pages 1941–1942.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Stephen Mussmann and Stefano Ermon. 2016. Learning and inference via maximum inner product search. In *International Conference on Machine Learning*, pages 2587–2596. PMLR.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, and Yinfei Yang. 2022. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Barlas Oguz, Kushal Lakhotia, Anchit Gupta, Patrick Lewis, Vladimir Karpukhin, Aleksandra Piktus, Xilun Chen, Sebastian Riedel, Scott Yih, Sonal Gupta, and Yashar Mehdad. 2022. Domain-matched pre-training tasks for dense retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1524–1534, Seattle, United States. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Devendra Sachan, Mostofa Patwary, Mohammad Shoeybi, Neel Kant, Wei Ping, William L. Hamilton, and Bryan Catanzaro. 2021. End-to-end training of neural retrievers for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6648–6662, Online. Association for Computational Linguistics.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Haitian Sun, William Cohen, and Ruslan Salakhutdinov. 2022. ConditionalQA: A complex reading comprehension dataset with conditional answers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3627–3637, Dublin, Ireland. Association for Computational Linguistics.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16:1–28.

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. 2018. Retrieval of the best counterargument without prior topic knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 241–251, Melbourne, Australia. Association for Computational Linguistics.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021a. TSDAE: Using transformer-based sequential denoising auto-encoderfor unsupervised sentence embedding learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 671–688, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. 2022. GPL: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2345–2360, Seattle, United States. Association for Computational Linguistics.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11897–11916, Bangkok, Thailand. Association for Computational Linguistics.

Shuai Wang, Shengyao Zhuang, and Guido Zuccon. 2021b. Bert-based dense retrievers require interpolation with bm25 for effective passage retrieval. In *Proceedings of the 2021 ACM SIGIR international conference on theory of information retrieval*, pages 317–324.

Zhiruo Wang, Jun Araki, Zhengbao Jiang, Md Rizwan Parvez, and Graham Neubig. 2023. Learning to filter context for retrieval-augmented generation. *arXiv preprint arXiv:2311.08377*.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, Copenhagen, Denmark. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv preprint*, abs/1910.03771.

Ji Xin, Chenyan Xiong, Ashwin Srinivasan, Ankita Sharma, Damien Jose, and Paul Bennett. 2022. Zero-shot dense retrieval with momentum adversarial domain invariant representations. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4008–4020, Dublin, Ireland. Association for Computational Linguistics.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, et al. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94.

Wen-tau Yih, Kristina Toutanova, John C. Platt, and Christopher Meek. 2011. Learning discriminative projections for text similarity measures. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 247–256, Portland, Oregon, USA. Association for Computational Linguistics.

Hongming Zhang, Xinran Zhao, and Yangqiu Song. 2021. A brief survey and comparative study of recent development of pronoun coreference resolution in English. In *Proceedings of the Fourth Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 1–11, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Appendix

A Datasets

A.1 General Setups

Different from the setup of the original dataset, we split one document into several chunks with a maximum of 128 words. This is because some dense retrievers such as DPR (Karpukhin et al., 2020) have the requirement of maximum input. Too long inputs will be overflow, leading to the loss of information. The chunk selected can be used to locate the document in the original dataset during the evaluation. Specifically, for SciQ, we reformulate the dataset from a QA task to a retrieval task. Originally, this task aims to answer scientific questions given the context. We collect the contexts in training, validation and test sets as the corpus.

Also, we will explain our motivation of focusing the queries containing subqueries:

- Chen et al. (2023) have studied the advantage of using propositions, i.e., the atomic units within documents, as the retrieval units for a complete query. And `MixGR` will not affect the retrieval results of single-subquery queries.
- In this work, we highlight the advantages of mixed-granularity retrieval that incorporates finer units in both queries and documents. Queries containing multiple subqueries are particularly well-suited to our research problem, as they will have different combinations with the documents.

A.2 Specific Setups

Here, we would specify the experimental setups for each dataset:

- BioASQ: In order to decrease the computational cost, we randomly sample the abstracts from the original corpus covered by BEIR (Thakur et al., 2021), which contains 14 914 603 abstracts.
- NFCorpus: The query that we apply here is the description version including training, validation, and test sets, instead of the keyword. These queries are not covered by BEIR (Thakur et al., 2021). Please refer to the original datasets of NFCorpus (Boteva et al., 2016).
- SciFact: We apply the dataset covered by BEIR (Thakur et al., 2021), using both training and test splits.
- SciDocs: We use the dataset covered by BEIR (Thakur et al., 2021), using the test split.
- SciQ: We use the queries in the test sets, and collect the contexts in training, validation, and test sets as the corpus.

For the documents (or propositions) of all the datasets, the format of the text is the concatenation of the title and the content.

Statistic	**BioASQ** (Tsatsaronis et al., 2015)	**NFCorpus** (Boteva et al., 2016)	**SciDocs** (Cohan et al., 2020)	**SciFact** (Wadden et al., 2020)	**SciQ** (Welbl et al., 2017)
#Query	3 743	1 016	1 000	1 109	884
#Multi-subquery queries	387	641	205	283	252
#Subqueries	821	3 337	522	614	874
#Documents	225 362	3 633	25 657	5 183	12 241
#Propositions	3 551 816	67 110	351 802	87 190	91 635

Table 4: Statistics for the BioASQ, NFCorpus, SciDocs, SciFact, and SciQ datasets. Please note that these statistics have been adjusted to exclude any obvious decomposition errors.

B Query and Document Decomposition

Here, we will complement the necessary information regarding the query and document decomposition.

B.1 Subquery and Proposition Examples

Here, we present examples of subqueries and propositions decomposed from the documents. The example is the decomposition of the example in Figure 1.

Query: Citrullinated proteins externalized in neutrophil extracellular traps act indirectly to perpetuate the inflammatory cycle via induction of autoantibodies.

- Subquery-0: Citrullinated proteins are externalized in neutrophil extracellular traps.
- Subquery-1: Citrullinated proteins act indirectly to perpetuate the inflammatory cycle.
- Subquery-2: The inflammatory cycle is perpetuated via induction of autoantibodies.

Document: RA sera and immunoglobulin fractions from RA patients with high levels of ACPA and/or rheumatoid factor significantly enhanced NETosis, and the NETs induced by these autoantibodies displayed distinct protein content. Indeed, during NETosis, neutrophils externalized the citrullinated autoantigens implicated in RA pathogenesis, and anti-citrullinated vimentin antibodies potently induced NET formation. Moreover, the inflammatory cytokines interleukin-17A (IL-17A) and tumor necrosis factor-α (TNF-α) induced NETosis in RA neutrophils. In turn, NETs significantly augmented inflammatory responses in RA and OA synovial fibroblasts, including induction of IL-6, IL-8, chemokines, and adhesion molecules. These observations implicate accelerated NETosis in RA pathogenesis, through externalization of citrullinated autoantigens and immunostimulatory molecules that may promote aberrant adaptive and innate immune responses in the joint and in the periphery, and perpetuate pathogenic mechanisms in this disease.

- Proposition-0: RA sera and immunoglobulin fractions from RA patients with high levels of ACPA and/or rheumatoid factor significantly enhanced NETosis.
- Proposition-1: NETs induced by these autoantibodies displayed distinct protein content.
- Proposition-2: During NETosis, neutrophils externalized the citrullinated autoantigens implicated in RA pathogenesis.
- Proposition-3: Anti-citrullinated vimentin antibodies potently induced NET formation.
- Proposition-4: Interleukin-17A (IL-17A) and tumor necrosis factor- (TNF-) induced NETosis in RA neutrophils.
- Proposition-5: NETs significantly augmented inflammatory responses in RA and OA synovial fibroblasts.
- Proposition-6: NETs inducing IL-6, IL-8, chemokines, and adhesion molecules occurred in RA and OA synovial fibroblasts.
- Proposition-7: These observations implicate accelerated NETosis in RA pathogenesis.
- Proposition-8: NETosis externalizes citrullinated autoantigens and immunostimulatory molecules.
- Proposition-9: NETosis may promote aberrant adaptive and innate immune responses in the joint and in the periphery.
- Proposition-10: NETosis may perpetuate pathogenic mechanisms in RA.

B.2 Remarks on *Propositioner*

During our manual check on the decomposition results of *propositioner* (Chen et al., 2023), we find the following potential flaws.

(1) Wrong logic during decomposition:

> *Query*: Identification of Design Elements for a Maturity Model for Interorganizational Integration: A Comparative Analysis
> $\rightarrow$ *Subqueries*: ['Identification of Design Elements for a Maturity Model for Interorganizational Integration.', 'A Comparative Analysis is used for identifying design elements.']

(2) Hallucination:

> *Query*: Bigger ocean waves and waves that carry more sediment cause a greater extent of what?
> $\rightarrow$ *Subqueries*: ['Bigger ocean waves cause a greater extent of erosion.', 'Waves that carry more sediment cause a greater extent of erosion.']

(3) Information loss:

> *Query*: The reduction was 1.6 ± 1.6 in controls. ...
> $\rightarrow$ *Subqueries*: ['The reduction in migraine headache was 1.6 1.6 in controls.', ...]

We find that the proposition will convert the questions to declarative sentences during decomposition. This may stem from the fact that its training corpus is Wikipedia, where a small portion of sentences are questions. Still, we find that *propositioner* can still decompose question-style queries, as shown in the following example:

> *Query*: What is the purpose of bright colors on a flower's petals?
> $\rightarrow$ *Subqueries*: ["The purpose of bright colors on a flower's petals is unknown."]

What is more, the propositioner may decompose the query to a sequence of single characters, but it is very rare: there are only four cases out of 4009 queries, i.e., around 0.1 % rate for this type of error.

B.3 Human Evaluation on Query and Document Decomposition

As mentioned in §3.1, we evaluate the decomposition outputs by *propositioner* (Chen et al., 2023), 100 samples for both query and document decomposition. Concretely, we ask three students at the post-graduate levels to evaluate the results, who are paid above the local minimum hourly wage. The instruction is shown below:

> Propositions in documents (or subqueries in queries) are defined as follows:
>
> - Each proposition conveys a distinct semantic unit, collectively expressing the complete meaning.
> - Propositions should be atomic and indivisible.
> - According to Choi et al. (2021), propositions should be contextualized and self-contained, including all necessary text information such as coreferences for clear interpretation.
>
> Given the document (query) and the corresponding propositions (subqueries) generated by the model, please check whether the document (query) has been correctly decomposed.
> Please write *1* as correct, and *0* as incorrect.

Model	`HuggingFace Checkpoint`
SimCSE (Gao et al., 2021)	princeton-nlp/unsup-simcse-bert-base-uncased
Contriever (Izacard et al., 2022)	facebook/contriever
DPR (Karpukhin et al., 2020)	facebook/dpr-ctx_encoder-multiset-base
	facebook/dpr-question_encoder-multiset-base
ANCE (Xiong et al., 2021)	castorini/ance-dpr-context-multi
	castorini/ance-dpr-question-multi
TAS-B (Hofstätter et al., 2021)	sentence-transformers/msmarco-distilbert-base-tas-b
GTR (Ni et al., 2022)	sentence-transformers/gtr-t5-base
GTE-Qwen2-1.5B-instruct (Li et al., 2023b)	Alibaba-NLP/gte-Qwen2-1.5B-instruct
E5-Mistral-7b-instruct (Wang et al., 2024)	intfloat/e5-mistral-7b-instruct

Table 5: Model checkpoints released on HuggingFace. For DPR and ANCE, two different models encode the context and query.

C Retrievers Models

Table 5 presents the dense retrievers applied in the experimental section, i.e., §4.

D Offline Indexing

The `pyserini` and `faiss` libraries were employed to convert retrieval units into embeddings. We leveraged GPUs for encoding these text units in batches with a batch size of 64 and a floating precision of f16. Following the preprocessing of these embeddings, all experiments conducted involved the utilization of an exact search method for inner products using `faiss.IndexFlatIP`,

E Searching Efficiency

Our search experiments utilized `pyserini` and were conducted on CPUs. Table 6 displays the time costs associated with six retrievers across five datasets for both query-document and subquery-proposition retrieval. We omitted two LLM-based retrievers from this analysis due to the challenges of executing search experiments without GPUs. Our findings indicate that the time ratio between subquery-proposition and query-chunk searches is approximately equal to the ratio of their respective sizes, subqueries versus queriess.

F Downstream Tasks

The templates of LLama for downstream QA tasks, i.e., SciFact and SciQ, are listed as follows. For SciQ, we convert it from multiple choice question answering to open question answering.

> Given the knowledge source: *context* \\n Question: *query* \\n Reply with one phrase. \\n Answer:

As SciFact is a fact-checking task, we here check whether LLMs can predict the relationship between the context and the claim. The template of SciFact is shown as follows:

> Context: {*context*} \\n Claim: {*query*} \\n For the claim, the context is supportive, contradictory, or not related? \\n Options: (A) Supportive (B) Contradictory (C) Not related \\n Answer:")

G Detailed Results

G.1 Ablation Study

As discussed in §6.1, we remove the component, i.e., query-doc similarity, query-prop similarity, or subquery-prop similarity, and assess the corresponding performance compared with `MixGR`. In Table 7, it is observed that `MixGR` outperforms all its components.

Dataset	Retriever	query-doc	subquery-prop	Ratio
BioASQ	SimCSE	0:44:33	1:55:41	2.60
	Contriever	0:44:17	1:49:24	2.47
	DPR	0:43:42	1:55:20	2.64
	ANCE	0:44:06	1:52:18	2.55
	TASB	0:25:04	0:59:17	2.37
	GTR	0:44:58	1:48:34	2.41
	Average	0:40:10	1:40:12	**2.51**
	#subquery / #query			**2.12**
NFCorpus	SimCSE	1:15:38	6:33:46	5.21
	Contriever	1:15:54	6:31:13	5.16
	DPR	1:17:12	6:49:53	5.31
	ANCE	1:17:25	6:35:25	5.11
	TASB	0:38:28	3:21:24	5.24
	GTR	1:13:16	6:17:00	5.15
	Average	1:01:06	5:34:54	**5.20**
	#subquery / #query			**5.21**
SciDocs	SimCSE	0:22:17	0:58:27	2.62
	Contriever	0:21:04	0:56:49	2.70
	DPR	0:20:42	0:55:43	2.69
	ANCE	0:20:52	0:57:42	2.77
	TASB	0:10:54	0:28:34	2.62
	GTR	0:22:26	0:54:37	2.43
	Average	0:19:59	0:51:55	**2.64**
	#subquery / #query			**2.55**
SciFact	SimCSE	0:32:24	1:11:04	2.19
	Contriever	0:32:00	1:08:48	2.15
	DPR	0:33:46	1:08:48	2.04
	ANCE	0:30:48	1:13:00	2.37
	TASB	0:17:01	0:38:09	2.24
	GTR	0:30:56	1:10:29	2.28
	Average	0:29:19	1:05:46	**2.21**
	#subquery / #query			**2.17**
SciQ	SimCSE	0:27:31	1:37:11	3.53
	Contriever	0:29:27	1:46:54	3.63
	DPR	0:27:06	1:36:49	3.57
	ANCE	0:28:02	1:38:13	3.50
	TASB	0:15:02	0:50:50	3.38
	GTR	0:27:49	1:31:52	3.30
	Average	**0:25:50**	1:30:18	**3.49**
	#subquery / #query			**3.47**

Table 6: Time cost for searching query-doc pairs and subquery-prop pairs with six retrievers on five datasets. The experiments run on 8 CPUs and 64 GB memory. The average time across the retrievers for each task is also highlighted. The ratio of searching time is quite close to the ratio of query size, subqueries versus queries.

Retriever	Setup	BioASQ		NFCorpus		SciDocs		SciFact		SciQ		Avg.	
		ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20
Unsupervised Dense Retrievers													
SimCSE	w/o $s_{s\text{-}p}$	25.0	26.1	19.6	16.0	8.7	11.5	32.3	37.0	76.1	78.0	32.3	33.7
	w/o $s_{q\text{-}p}$	27.6	28.6	21.4	17.4	8.5	11.6	33.1	37.4	77.9	79.6	33.7	34.9
	w/o $s_{q\text{-}d}$	32.4	32.4	22.8	18.6	8.5	11.9	33.9	39.0	80.7	82.2	35.7	36.8
	MixGR	30.7	31.3	22.3	18.1	9.1	12.2	34.8	39.8	84.0	85.5	36.2	37.4
Contriever	w/o $s_{s\text{-}p}$	66.5	70.5	43.6	36.2	14.8	20.0	65.6	69.9	78.0	80.1	53.7	55.3
	w/o $s_{q\text{-}p}$	66.9	71.2	43.0	36.6	14.6	20.1	66.3	70.8	81.6	83.3	54.5	56.4
	w/o $s_{q\text{-}d}$	66.1	70.4	43.2	36.3	14.7	20.0	65.0	69.5	83.3	84.8	54.5	56.2
	MixGR	67.0	71.7	44.0	37.1	15.5	20.7	66.4	71.0	85.2	86.7	55.6	57.5
Supervised Dense Retrievers													
DPR	w/o $s_{s\text{-}p}$	43.0	44.3	26.5	21.9	8.2	11.2	35.0	40.8	66.6	69.9	35.9	37.6
	w/o $s_{q\text{-}p}$	42.9	44.5	27.5	22.8	7.5	11.2	38.3	42.4	71.0	73.1	37.5	38.8
	w/o $s_{q\text{-}d}$	44.0	45.0	26.6	22.2	8.0	11.2	38.0	42.1	69.5	72.2	37.2	38.5
	MixGR	44.6	45.8	27.7	22.9	8.2	11.5	39.4	43.6	73.6	76.1	38.7	40.0
ANCE	w/o $s_{s\text{-}p}$	52.7	52.9	30.7	25.2	10.0	13.7	45.8	48.9	69.0	72.0	41.7	42.6
	w/o $s_{q\text{-}p}$	51.4	52.8	32.0	26.2	9.0	13.4	46.8	50.4	71.3	73.9	42.1	43.3
	w/o $s_{q\text{-}d}$	52.9	54.1	30.8	25.1	8.8	13.4	44.9	48.6	67.8	70.1	41.0	42.3
	MixGR	53.4	54.7	31.9	25.9	9.6	14.1	46.8	49.9	74.4	76.8	43.2	44.3
TAS-B	w/o $s_{s\text{-}p}$	71.3	73.2	42.9	34.7	13.8	19.2	61.4	66.7	86.7	87.0	55.2	56.2
	w/o $s_{q\text{-}p}$	70.1	72.6	42.9	34.9	13.8	19.6	63.2	67.3	88.3	88.8	55.7	56.7
	w/o $s_{q\text{-}d}$	70.0	72.6	42.7	34.5	13.6	18.8	62.1	65.3	85.2	85.9	54.8	55.4
	MixGR	71.7	74.0	43.6	35.2	14.0	19.6	62.7	66.9	90.5	91.0	56.5	57.3
GTR	w/o $s_{s\text{-}p}$	66.5	67.8	43.2	35.2	13.4	18.9	60.9	64.5	87.2	87.5	54.2	54.8
	w/o $s_{q\text{-}p}$	65.4	67.0	43.0	35.5	13.8	19.5	60.6	64.7	88.4	88.5	54.2	55.0
	w/o $s_{q\text{-}d}$	66.2	68.0	42.4	34.9	12.6	18.0	61.5	64.4	89.0	89.3	54.3	54.9
	MixGR	66.8	68.6	43.3	35.6	13.6	19.2	60.9	64.5	92.9	93.0	55.5	56.2
LLM-based Retrievers													
GTE-Qwen	w/o $s_{s\text{-}p}$	64.1	67.8	44.0	36.4	16.2	22.5	68.4	72.6	81.4	82.2	54.8	56.3
	w/o $s_{q\text{-}p}$	61.0	64.9	43.0	36.0	15.4	21.0	66.7	71.4	82.6	83.2	53.7	55.3
	w/o $s_{q\text{-}d}$	67.5	71.4	45.5	37.4	17.5	24.7	70.0	73.5	83.3	84.3	56.7	58.3
	MixGR	66.5	70.1	44.9	37.5	17.2	23.5	70.3	74.3	86.7	87.5	57.1	58.6
E5-Mistral	$s_{q\text{-}d}$	75.7	79.6	50.6	42.2	18.1	25.0	76.0	78.6	88.4	88.6	61.8	62.8
	$s_{q\text{-}p}$	73.9	78.4	49.9	41.6	17.9	24.5	75.6	78.4	89.7	90.1	61.4	62.6
	$s_{s\text{-}p}$	74.9	79.2	48.8	40.8	17.1	23.4	74.7	77.4	85.5	86.8	60.2	61.5
	MixGR	75.9	79.8	50.4	42.0	18.1	24.9	76.5	79.2	90.7	91.2	62.3	63.4

Table 7: Ablation study (nDCG@k = 5, 20 in percentage, abbreviated as ND@k): We evaluated five distinct scientific retrieval datasets using two unsupervised, four supervised, and two LLM-based retrievers. The retrieval results were compared using various metrics: MixGR w/o $s_{s\text{-}q}$, MixGR w/o $s_{q\text{-}p}$, MixGR w/o $s_{s\text{-}p}$, and MixGR, as detailed in §3.2.

G.2 MixGR for Propositional Retrieval

Here, we evaluate MixGR on the retrieval units beyond documents, e.g., propositions, which Table 8 present. We observe that MixGR can outperform the previous document retrieval based on the similarity between query and proposition, on proposition retrieval, as discussed in §6.3.

G.3 Advantageous pattern for finer granularity measurement

In Table 10, we can notice the average number of propositions in $r_{q\text{-}d} \prec r_{s\text{-}p}$ is more than $r_{q\text{-}d} \succ r_{s\text{-}p}$. This shows that the finer granularity can better deal with the documents with more propositions than the original query-document similarity.

G.4 Document retrieval for queries containing only a single subquery

In this experiment, we demonstrate the impact of MixGR on queries that contain only a single subquery. Unlike queries with multiple subqueries, MixGR omits the similarity measurement from the perspectives of subqueries and propositions, $s_{s\text{-}p}$, during RRF. The result, presented in Table 9, includes the result of document retrieval for the queries including either one single or multiple subqueries and illustrates the effectiveness of MixGR across a general query format.

	Setup	SciFact EM@50	SciFact EM@200	SciQ EM@50	SciQ EM@200
Unsupervised Dense Retrievers					
SimCSE	$s_{q\text{-}d}$	42.4$_{\pm 3.8}$	59.9$_{\pm 3.3}$	56.5$_{\pm 0.2}$	61.4$_{\pm 0.5}$
	MixGR	**44.5**$_{\pm 3.8}$	**61.8**$_{\pm 1.5}$	**58.5**$_{\pm 0.4}$	**63.0**$_{\pm 0.2}$
Contriever	$s_{q\text{-}d}$	**46.7**$_{\pm 4.1}$	62.8$_{\pm 3.8}$	56.9$_{\pm 0.5}$	**63.7**$_{\pm 0.6}$
	MixGR	45.2$_{\pm 4.6}$	**67.8**$_{\pm 5.1}$	**57.7**$_{\pm 0.5}$	63.2$_{\pm 0.5}$
Supervised Dense Retrievers					
DPR	$s_{q\text{-}d}$	46.7$_{\pm 2.4}$	56.6$_{\pm 2.7}$	55.1$_{\pm 0.3}$	60.1$_{\pm 0.5}$
	MixGR	**48.6**$_{\pm 2.4}$	**62.0**$_{\pm 1.8}$	**58.6**$_{\pm 0.3}$	**60.9**$_{\pm 0.0}$
ANCE	$s_{q\text{-}d}$	**47.3**$_{\pm 1.9}$	61.3$_{\pm 2.0}$	53.9$_{\pm 0.3}$	**60.5**$_{\pm 0.3}$
	MixGR	43.4$_{\pm 2.6}$	**64.3**$_{\pm 2.2}$	**54.7**$_{\pm 0.3}$	59.9$_{\pm 0.4}$
TAS-B	$s_{q\text{-}d}$	45.5$_{\pm 4.8}$	**66.3**$_{\pm 4.1}$	56.2$_{\pm 0.3}$	60.8$_{\pm 0.2}$
	MixGR	**47.1**$_{\pm 4.1}$	65.9$_{\pm 3.6}$	**58.1**$_{\pm 0.2}$	**62.8**$_{\pm 0.2}$
GTR	$s_{q\text{-}d}$	39.5$_{\pm 3.1}$	**66.7**$_{\pm 3.4}$	59.8$_{\pm 0.3}$	64.5$_{\pm 0.0}$
	MixGR	**43.6**$_{\pm 4.1}$	62.4$_{\pm 4.0}$	**60.5**$_{\pm 0.0}$	**65.1**$_{\pm 0.2}$
LLM-based Retrievers					
GTE-Qwen	$s_{q\text{-}d}$	**47.8**$_{\pm 5.8}$	64.9$_{\pm 2.6}$	**59.7**$_{\pm 0.2}$	**65.1**$_{\pm 0.4}$
	MixGR	42.8$_{\pm 5.2}$	**69.8**$_{\pm 3.3}$	58.2$_{\pm 0.0}$	63.4$_{\pm 0.2}$
E5-Mistral	$s_{q\text{-}d}$	**44.6**$_{\pm 4.4}$	63.0$_{\pm 3.5}$	55.4$_{\pm 0.4}$	62.9$_{\pm 0.0}$
	MixGR	42.8$_{\pm 3.4}$	**63.6**$_{\pm 2.2}$	**58.5**$_{\pm 0.2}$	**63.3**$_{\pm 0.3}$

Table 8: Scientific Question Answering (Exact Match) was conducted using LLama-3 (Touvron et al., 2023) with propositions retrieved by eight retrievers. Here, EM@50 and EM@200 have been reported, where the first 50 and 200 words are fed into the reader models. **Bold** indicates superior performance, and it is observed that retrieval using MixGR on proposition units generally outperforms the baseline. Numbers following ± present standard deviations of EM, generated with temperature 0.1 and top_p 0.7 on three seeds.

H MixGR for Other Domains

Our work provides a comprehensive analysis of the impact of MixGR on scientific text retrieval, considering both the variety of datasets and the use of dense retrievers. The applicability of MixGR to other domains remains an open question. We explore this by conducting document retrieval experiments on three distinct datasets: ConditionalQA (Sun et al., 2022), FiQA (Maia et al., 2018), and Arguana (Wachsmuth et al., 2018), which belong to the domains of law, finance, and argumentation, respectively. The results are detailed in Table 11. We observe that MixGR's benefits are considerably more limited, or even negative, outside the scientific context. This disparity may be attributed to the varying degrees of alignment between the domain-specific characteristics of each field and the training corpus of the dense retrievers. Or, *propositioner* can not perform well in these domains. Such findings further underscore the potentially distinct domain-specific nature of scientific document retrieval.

I Licences of Scientific Artifacts

Retriever	Setup	BioASQ		NFCorpus		SciDocs		SciFact		SciQ		Avg.	
		ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20
Unsupervised Dense Retrievers													
SimCSE	$s_{q\text{-}d}$	17.7	17.9	14.5	12.1	27.9	31.9	4.9	7.1	50.8	56.5	23.2	25.1
	`MixGR`	**26.2**	**27.1**	**19.8**	**16.2**	**34.8**	**39.1**	**6.7**	**9.5**	**70.1**	**73.0**	**31.5**	**33.0**
Contriever	$s_{q\text{-}d}$	62.0	64.4	39.9	33.1	66.0	69.1	12.4	17.2	56.4	60.5	47.3	48.9
	`MixGR`	**64.5**	**67.7**	**41.4**	**34.7**	**67.0**	**70.7**	**13.2**	**18.5**	**75.3**	**77.7**	**52.3**	**53.9**
Supervised Dense Retrievers													
DPR	$s_{q\text{-}d}$	36.0	35.9	23.8	19.7	35.9	40.5	6.0	8.8	51.2	56.3	30.6	32.2
	`MixGR`	**41.0**	**41.6**	**26.1**	**21.5**	**40.0**	**44.1**	**6.9**	**9.9**	**63.2**	**67.0**	**35.4**	**36.8**
ANCE	$s_{q\text{-}d}$	43.4	43.3	28.4	23.2	43.2	47.1	7.8	11.0	56.8	61.2	35.9	37.1
	`MixGR`	**48.4**	**48.9**	**30.5**	24.6	**46.3**	**50.1**	**8.6**	**11.9**	**65.2**	**68.7**	**39.8**	**40.9**
TAS-B	$s_{q\text{-}d}$	66.4	67.6	39.6	31.9	62.5	66.2	12.1	16.5	75.4	77.7	51.2	52.0
	`MixGR`	**69.2**	**71.1**	**40.8**	**32.9**	**64.1**	**67.7**	**12.5**	**16.8**	**81.2**	**82.7**	**53.6**	**54.2**
GTR	$s_{q\text{-}d}$	61.6	61.9	39.9	32.3	58.8	62.3	12.0	16.2	72.6	75.6	49.0	49.7
	`MixGR`	**65.5**	**66.8**	**41.0**	**33.5**	**62.0**	**65.3**	**12.2**	**16.7**	**82.4**	**83.4**	**52.6**	**53.1**
LLM-based Retrievers													
GTE-Qwen	$s_{q\text{-}d}$	53.0	54.0	38.2	31.3	62.8	67.2	10.0	14.1	64.0	67.8	45.6	46.9
	`MixGR`	**63.5**	**66.6**	**42.3**	**35.2**	**69.5**	**73.2**	**13.6**	**19.1**	**76.6**	**78.7**	**53.1**	**54.6**
E5-Mistral	$s_{q\text{-}d}$	73.3	75.4	47.2	39.0	74.7	77.8	16.5	22.6	79.7	81.7	58.3	59.3
	`MixGR`	**76.7**	**79.3**	**47.4**	**39.3**	**75.8**	**79.0**	**16.0**	**22.4**	**83.9**	**84.9**	**60.0**	**61.0**

Table 9: Document Retrieval Performance (nDCG@k = 5, 20 in percentage, abbreviated as ND@k) on all the queries, including the ones containing single and multiple subqueries. The setup is similar to Table 2. comparing the retrieval results based on $s_{q\text{-}d}$ and `MixGR`. **Bold** presents the best performance across the metrics. We can notice that `MixGR` outperform the previous query-document retrieval for each entry.

Model	Avg. #prop in $r_{q\text{-}d} \prec r_{s\text{-}p}$	Avg. #prop in $r_{q\text{-}d} \succ r_{s\text{-}p}$
SimCSE	9.06	6.32
Contriever	8.25	7.24
ANCE	8.12	8.15
DPR	8.54	7.88
GTR	8.45	6.79
TAS-B	8.00	7.52
GTE-Qwen	9.31	5.79
E5-Mistral	8.78	6.73

Table 10: Average number of propositions in two sets of document for different retrievers, i.e., $r_{q\text{-}d} \prec r_{s\text{-}p}$ and $r_{q\text{-}d} \succ r_{s\text{-}p}$. We can notice the average number of propositions in $r_{q\text{-}d} \prec r_{s\text{-}p}$ is more than $r_{q\text{-}d} \succ r_{s\text{-}p}$. This shows that the finer granularity can better deal with the documents with more propositions.

Retriever	Setup	Arguana		ConditionalQA		FiQA		Avg.	
		ND@5	ND@20	ND@5	ND@20	ND@5	ND@20	ND@5	ND@20
Unsupervised Dense Retrievers									
SimCSE	$s_{q\text{-}d}$	16.4	25.9	52.3	58.0	8.4	10.9	25.7	31.6
	$s_{q\text{-}p}$	12.5	20.9	53.7	59.5	7.6	9.7	24.6	30.0
	$s_{s\text{-}p}$	6.3	12.3	42.8	50.8	9.3	11.6	19.5	24.9
	`MixGR`	12.7	22.4	57.7	63.3	10.6	13.8	27.0	33.2
Contriever	$s_{q\text{-}d}$	25.9	36.0	82.5	83.9	25.0	29.9	44.5	49.9
	$s_{q\text{-}p}$	24.8	35.9	81.8	83.5	18.8	23.1	41.8	47.5
	$s_{s\text{-}p}$	24.1	34.5	63.3	67.2	18.6	22.9	35.3	41.5
	`MixGR`	28.7	39.2	83.5	84.5	24.7	29.8	45.6	51.2
Supervised Dense Retrievers									
DPR	$s_{q\text{-}d}$	9.0	16.6	58.5	63.6	12.0	14.6	26.5	31.6
	$s_{q\text{-}p}$	8.4	16.9	60.1	64.7	8.4	10.9	25.6	30.8
	$s_{s\text{-}p}$	6.1	12.2	34.8	41.8	9.2	11.8	16.7	21.9
	`MixGR`	8.2	16.3	59.9	65.4	11.2	14.9	26.4	32.2
ANCE	$s_{q\text{-}d}$	12.0	20.5	64.2	68.0	14.6	18.2	30.3	35.6
	$s_{q\text{-}p}$	11.7	21.3	64.0	68.2	8.5	10.9	28.1	33.5
	$s_{s\text{-}p}$	10.1	18.6	41.4	48.1	8.4	11.3	20.0	26.0
	`MixGR`	12.4	21.8	66.2	69.8	12.8	16.2	30.5	36.0
TAS-B	$s_{q\text{-}d}$	27.9	37.8	75.3	77.9	26.7	31.5	43.3	49.0
	$s_{q\text{-}p}$	18.8	30.5	76.4	78.7	15.3	19.7	36.8	43.0
	$s_{s\text{-}p}$	12.9	20.8	60.8	65.2	13.9	17.8	29.2	34.6
	`MixGR`	22.6	33.6	77.7	79.2	22.8	27.9	41.1	46.9
GTR	$s_{q\text{-}d}$	31.4	40.7	79.8	82.3	34.4	39.6	48.5	54.2
	$s_{q\text{-}p}$	25.6	36.9	80.1	82.0	22.8	27.4	42.8	48.8
	$s_{s\text{-}p}$	20.4	30.0	62.9	67.7	19.6	24.2	34.3	40.6
	`MixGR`	29.4	39.4	82.4	84.1	30.8	36.1	47.5	53.2

Table 11: Comparison between `MixGR` and its components on ConditionalQA, Arguana, and FiQA. We can find that the similarity based on the finer granularity $s_{s\text{-}p}$ and `MixGR` won't bring as many benefits as their performance in the scientific domains, even the degradation.

Artifacts/Packages	Citation	Link	License
Artifacts(datasets/benchmarks).			
SciFact	(Wadden et al., 2020)	`https://huggingface.co/datasets/BeIR/scifact`	cc-by-sa-4.0
SciDocs	(Cohan et al., 2020)	`https://huggingface.co/datasets/BeIR/scidocs`	cc-by-sa-4.0
SciQ	(Welbl et al., 2017)	`https://huggingface.co/datasets/bigbio/sciq`	cc-by-nc-3.9
NFCorpus	(Boteva et al., 2016)	`https://huggingface.co/datasets/BeIR/nfcorpus`	cc-by-sa-4.0
Packages			
PyTorch	(Paszke et al., 2019)	`https://pytorch.org/`	BSD-3 License
transformers	(Wolf et al., 2019)	`https://huggingface.co/transformers/v2.11.0/index.html`	Apache License 2.0
numpy	(Harris et al., 2020)	`https://numpy.org/`	BSD License
matplotlib	(Hunter, 2007)	`https://matplotlib.org/`	BSD compatible License
vllm	(Kwon et al., 2023)	`https://github.com/vllm-project/vllm`	Apache License 2.0
Models			
LLaMA-3	(Touvron et al., 2023)	`https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct`	LICENSE
SimCSE	(Gao et al., 2021)	`https://huggingface.co/princeton-nlp/unsup-simcse-bert-base-uncased`	MIT license
Contriever	(Izacard et al., 2022)	`https://huggingface.co/facebook/contriever`	License
DPR	(Karpukhin et al., 2020)	`https://huggingface.co/facebook/dpr-ctx_encoder-multiset-base`	cc-by-nc-4.0
ANCE	(Xiong et al., 2021)	`https://huggingface.co/castorini/ance-dpr-context-multi`	MIT license
TAS-B	(Hofstätter et al., 2021)	`https://huggingface.co/sentence-transformers/msmarco-distilbert-base-tas-b`	Apache License 2.0
GTR	(Ni et al., 2022)	`https://huggingface.co/sentence-transformers/gtr-t5-base`	Apache License 2.0

Table 12: Details of datasets, major packages, and existing models we use. The datasets we reconstructed or revised and the code/software we provide are under the MIT License.